Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID WHEEL SLIP

Applicants: Jiang et al Page 1 of 8



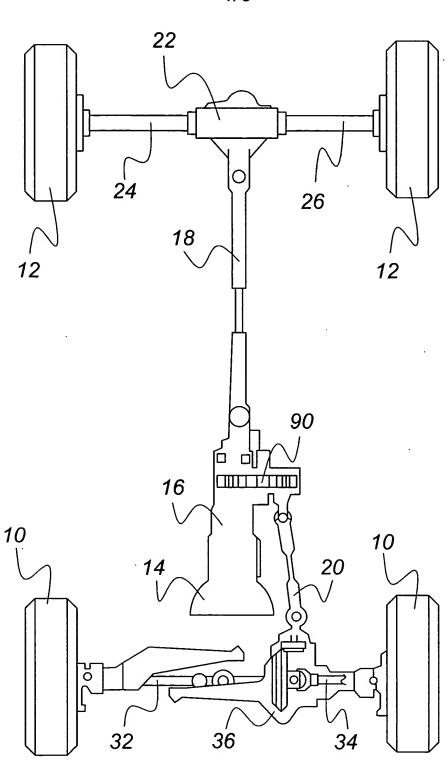


Figure 1

Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP
Applicants: Jiang et al
Page 2 of 8

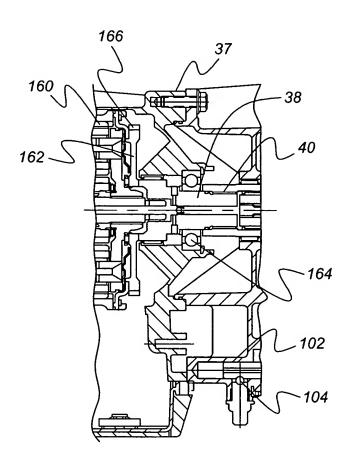
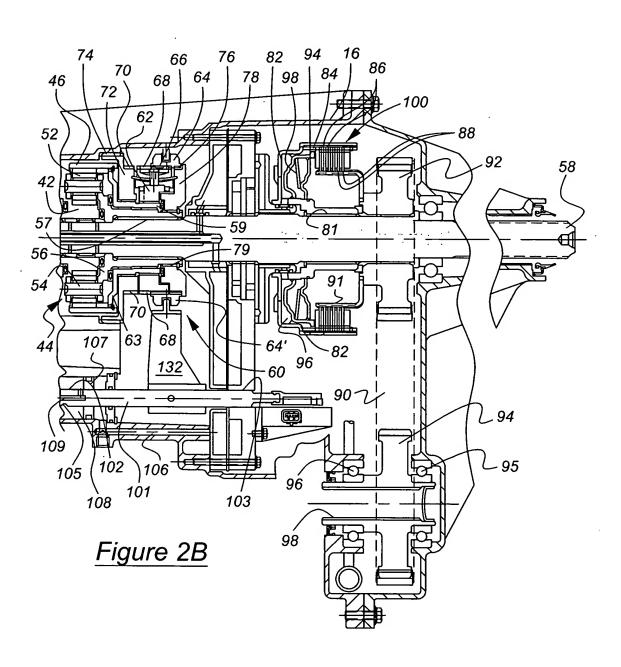


Figure 2A

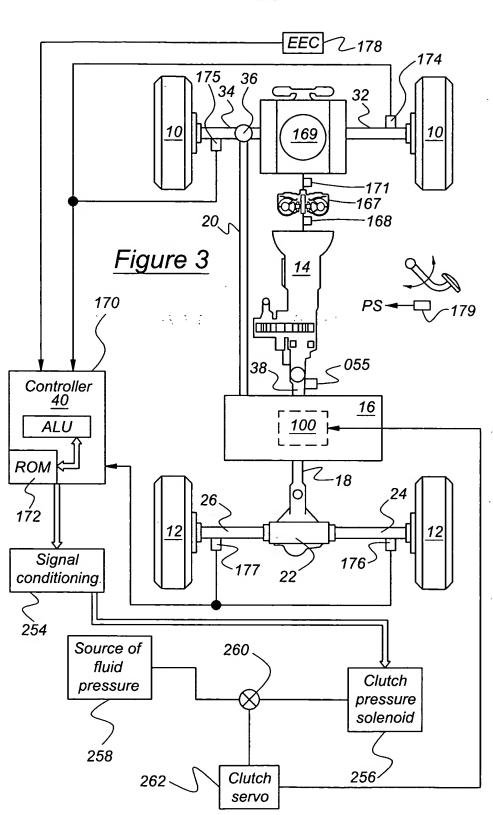
Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 3 of 8



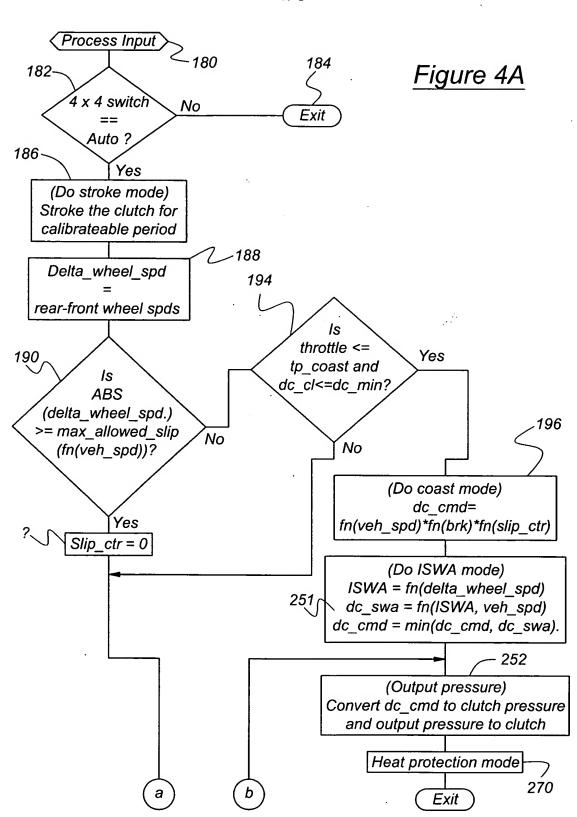
Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 4 of 8



Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 5 of 8



Title: METHOD AND SYSTEM FOR CONTROLLING

A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 6 of 8

6/8

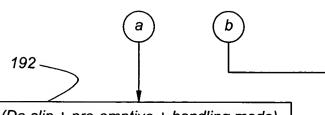


Figure 4B

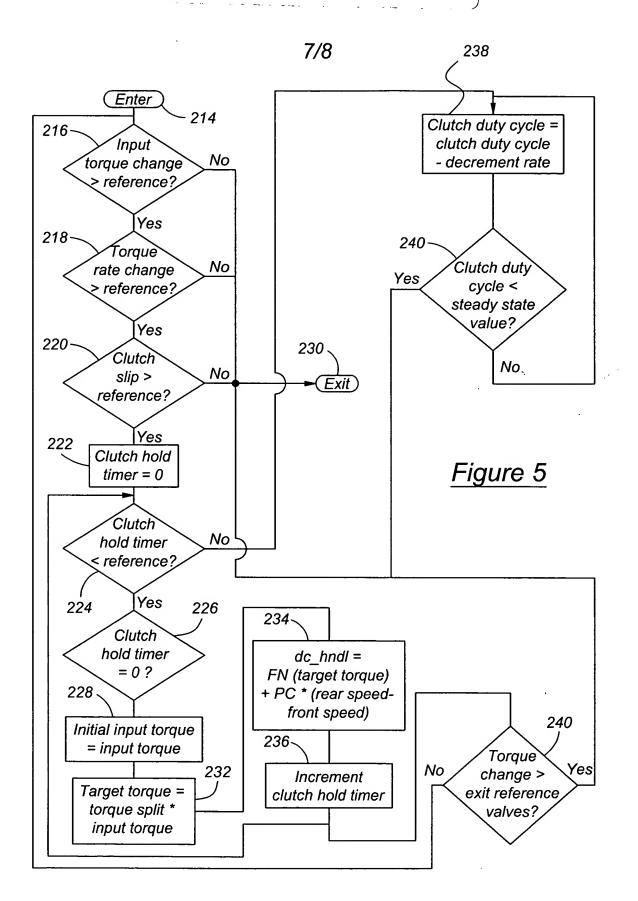
(Do slip + pre-emptive + handling mode) traction

Increment slip_ctr counter Err = delta_wheel_spd-fn(veh_spd). $kp,ki = fn(throttle,veh_spd)n*fn(err).$ kc = fn (delta_wheel_spd)

- 1. TP based pre-emptive duty cycle. dc_pps = fn(veh_spd, throttle) * fn(slip_ctr).
 - 2. Handling duty cycle.
 - 3. PI close loop duty cycle. dc cl = (err * kp + err + ki) * kc
- 4. TP rate based pre emptive duty cycle. dc_pre = fn(throttle rate, veh_spd). hold dc_pre constant for a while and then decrement every loop.
- 5. Total command duty cycle. dc_cmd = greater of (dc_pps, dc_hndl) + dc_cl + dc_pre.

Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 7 of 8



Title: METHOD AND SYSTEM FOR CONTROLLING A TRANSFER CASE CLUTCH TO AVOID

WHEEL SLIP Applicants: Jiang et al Page 8 of 8

